

ĐỀ THI MẪU CUỐI KỲ MÔN KỸ THUẬT LẬP TRÌNH

(Gồm 3 phần: Basics – Recursion and Pointer – OOP kèm 2 câu tự luận)

Lưu ý: các đáp án điền vào không có dấu cách ở bất cứ đâu, trừ trường hợp đề yêu cầu

PART 1: BASIC (10 câu)

1. Cú pháp khai báo biến nào sau đây là **không** hợp lệ?

- A. _long
- B. short**
- C. myNum
- D. a8bi

2. Cho đoạn code sau:

```
string name;
```

```
cin >> name;
```

```
cout << "Your name is: " << name;
```

Khi người dùng nhập vào: Nguyen Van A, kết quả in ra màn hình là?

- A. Your name is Nguyen Van A**
- B. Your name is Nguyen
- C. Your name is A
- D. Your name is Nguyen Van

3. Cho câu lệnh sau:

```
float a[5] = {1.0, 2.0, 3.0, 4.0, 5.0};
```

Câu lệnh nào sau đây dùng để in ra giá trị của phần tử đầu tiên của mảng a?

- A. cout << a << '\n';
- B. cout << a[0] << '\n';**
- C. cout << &a[0] << '\n';
- D. Không có câu nào đúng

4. Cho đoạn code sau:

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
    string s = "microsoft";
    cout << s.substr(0, 2);
}
```

lay tu index, do dai la len

Kết quả khi in ra màn hình là?

Điền câu trả lời ở đây: [mi](#)

5. Cho đoạn code sau:

```
int a[5] = {1,2,3,4,5};
for (int i = 0; i < 5; i += 2)
{
    a[i]--;
    cout << a[i--];
}
```

a0 = 0
0
i = -1
i = 1
a1 = 1
1
i = 0

Kết quả khi chạy chương trình là?

Điền câu trả lời ở đây (nếu như chương trình có lỗi, hãy chỉ ra dòng lệnh bị lỗi): [01234](#)

6. Cho đoạn code sau:

```
#define sub(a,b) a-b
cout << 3*sub(5*2,3*3) + 1;
```

Kết quả của chương trình là?

A. 4

B. -2

C. 0

D. Chương trình có lỗi

7. Cho đoạn code sau:

```
double pi = 3.1416;  
cout << '*' << setw(7) << setprecision(7) << pi << "*" \n";
```

Kết quả của chương trình là?

A. *3.1416*

B. * 3.1416*

C. *3.1416000*

D. * 3.1416000*

8. Cho đoạn code sau:

```
int s = 1, i = -3;  
while(i++)  
{  
    s -= i;  
}  
cout << s + (2 >> --i);
```

Khi thay đổi dòng `while(i++)` thành `while(++i)` thì kết quả của chương trình lúc sau là?

Điền câu trả lời vào đây (nếu như không có sự thay đổi output của chương trình, ghi `no_change`, nếu chương trình gặp lỗi, hãy chỉ ra dòng lệnh bị lỗi): 2

9. Cho đoạn code sau:

```
for (int i = 0; i < 10; 1i++) // line 1  
{  
    3if (++i > 2) break; // line 2  
    1  
}
```

Khẳng định đúng là:

- A. Lỗi ở line 1
- B. Chương trình lặp 2 lần**
- C. Lỗi ở line 2
- D. Chương trình lặp 3 lần

10. Khi nói về string, khẳng định nào sau đây là đúng?

- A. String có độ dài cố định
- B. Một object có kiểu string luôn kết thúc bằng ký tự '\0'**
- C. String có độ dài động
- D. Không thể truy xuất phần tử bất kỳ trong string

PART 2: Recursion and Pointer (10 câu)

11. Cho đoạn chương trình dùng để tính giai thừa của 1 số n như sau:

```
typedef unsigned long long int ull;  
ull foo (int n)
```

```
{  
    if (n == 0)  
    {  
        return 1;  
    }  
    return n * foo(n--);  
}
```


Khẳng định đúng khi thực hiện lệnh foo(5) là?

- A. Chương trình không có lỗi, kết quả là 120
- B. Chương trình không có lỗi, kết quả là 0
- C. Chương trình không chạy được, bị lỗi Stack Overflow**
- D. Cả 3 câu trên đều sai

12. Cho hàm đệ quy sau:

```
int foo(int a, int b)
{
    static int m = 0; // Line 0
    if (m % a == 0 && m % b == 0)
    {
        return m;
    }
    m += b;
    return foo(a, b);
}
```

Biết rằng khi thực hiện lệnh `foo(5, 4)` thì kết quả là 20. Bỏ từ khóa “static” ở Line 0 đi và thực hiện lại lệnh `foo(5, 4)`, khẳng định nào sau đây **không** đúng?

- A. Chương trình bị lỗi Stack Overflow
- B. Từ khóa “static” ở Line 0 giúp cho giá trị của biến được lưu lại và sử dụng cho các lần gọi hàm sau
- C. Có thể khai báo `int m = 0` ở ngoài hàm `foo` (biến cục bộ), trong hàm `foo` đưa dòng `m += b` lên thay thế vị trí của Line 0 thì chương trình vẫn in ra giá trị 20
-  D. Cả 3 câu đều sai

 13. Cho đoạn code sau:

```
#include <bits/stdc++.h>
using namespace std;
int main()
{
    string s = "microsoft";
    for (auto it = s.rbegin(); it != s.rend(); ++it)
    {
        khac \0
    }
}
```

```

        cout << *it;
    }
}

```

Kết quả của đoạn code trên là?

Điền kết quả vào đây: tfosorcim

14. Cho mảng A có 16 phần tử là số nguyên (mỗi số nguyên có kích thước 4 byte). Nếu địa chỉ của phần tử thứ 7 của A là X thì địa chỉ của phần tử đầu tiên của mảng là?

Điền câu trả lời vào đây:
 thu 6 : x-4
 thu 5 : x-4*2
 ...
 thu 1 : x-4*6 = x-24

15. Cho đoạn code sau:

```

double foo(double* pre, double* curr)
{
    double S = 0;
    while(pre < curr)
    {
        S += (*pre) * (*pre);
        pre++;
    }
    return sqrt(S);
}

```

Với dòng lệnh khởi tạo mảng: `double A[5] = {1.0, 2.0, 4.0, 2.0, 1.0}`, khẳng định nào sau đây **không** đúng?

- A. `foo(&A[0], &A[4])` cho kết quả là 5.00 ✓
- B.** `foo(&A[3], NULL)` cho kết quả là 4.58
- C. `foo(A + 2, A + 5)` cho kết quả là 4.58 ✓
- D. không thể thực hiện `foo(NULL, A + 1)` ✓

16. Cho đoạn code sau:

int q = 10; //0 ✓

const int* p = &q; //1 ✓

int q2 = 12; //2 ✓

*p = 7; //3 ✗

p = &q2; //4 ✓

Cho các khẳng định sau:

- 1) Chỉ có dòng lệnh số 3 bị lỗi ✓
- 2) p có thể thay đổi được địa chỉ, nhưng không thể thay đổi được giá trị lưu ở địa chỉ đó ✓
- 3) Chỉ có dòng lệnh số 4 bị lỗi ✗
- 4) Chương trình vẫn chạy bình thường ✗

Những khẳng định **sai** là?

Điền câu trả lời vào đây:

3 4

17. Cho đoạn code sau:

int a = 20, b = 5;

int *p = &a, *q = &b;

p = q; →

Cho các khẳng định sau:

- 1) p trỏ đến a ✗
- 2) q trỏ đến b ✓
- 3) Khi thay đổi giá trị của b thì p không thay đổi ✗
- 4) p và q đều đang trỏ đến b ✓
- 5) Khi delete p hoặc q thì con trỏ còn lại cũng mất đi ✓

Số khẳng định đúng là:

A. 1

B. 2

C. 3

D. 4

18. Cho đoạn code sau:

```
struct node
{
    int data;
    node* next;
}

void fun(node* head)
{
    if (head == nullptr)
    {
        return;
    }
    cout << head->data << " ";
    if (head->next != nullptr)
    {
        fun(head->next->next);
    }
    cout << head->data << " ";
}
```

Thực hiện lệnh fun với list là $2 \rightarrow 4 \rightarrow 6 \rightarrow 8 \rightarrow 10 \rightarrow 12$, kết quả thu được là?

A. 2 6 10 6 2

B. 2 4 6 8 10 12 10 8 6 4 2

C. 2 6 10 10 6 2

D. Cả 3 câu trên sai

19. Cho đoạn code sau:

```
int foo(int a, int b)
{
    if (b == 0)
    {
        return 1;
    }
    else if (!(b & 1))
    {
        return foo(a, b / 2) * foo(a, b / 2);
    }
}
```



```

    }
    else
    {
        return a * foo(a, (b - 1) / 2) * foo(a, (b - 1) / 2);
    }
}

```

foo(5, 4) trả về kết quả là?

Điền câu trả lời vào đây: 625

20. Cho đoạn code sau:

```

double foo(double x, unsigned long long int y)
{
    if (!y)
    {
        return x;
    }
    return foo(x / (y << 1), y / 2);
}

```

Khi thực hiện lệnh `cout << fixed << setprecision(1) << foo(5, 5)`, kết quả là?

A. 0.1

B. Chương trình bị lỗi

C. 0.2

D. 0.0

PART 3: OOP (10 câu)

21. Khẳng định nào sau đây là **sai** khi nói về constructor?

A. Constructor có ở cả struct và class

B. Một class có thể có nhiều constructor khác nhau

C. Khi đã khai báo 1 constructor có tham số, trình biên dịch vẫn cung cấp constructor mặc định (không có tham số) khi khởi tạo object mà không truyền tham số

D. Constructor có tên trùng với tên của class

22. Khi nói về quyền truy cập các thuộc tính của struct và class, khẳng định nào sau đây là **sai**?

- A. Quyền truy cập của struct luôn là public, còn của class có thể thay đổi tùy ý
- B. Trong class, khi không chỉ định quyền truy cập thì class giống như struct**
- C. Các thuộc tính của class chỉ có thể truy cập bên ngoài class khi có quyền truy cập public, còn struct có thể truy cập ở bất cứ đâu trong chương trình
- D. Các thuộc tính private của class A chỉ có thể truy cập từ class B nếu A có friend class là B

23. Cho đoạn code sau:

```
template<typename T1, typename T2>
```

```
T1 add(T1 a, T2 b)
```

```
{
```

```
    return a + b;
```

```
}
```

add(3, 4.5) cho kết quả là?

A. 7.5

B. 7.0

C. 7

D. Chương trình bị lỗi

24. Cho đoạn code sau:

```
class Complex
```

```
{
```

```
public:
```

```
    int real;
```

```
    int im;
```

```
public:
```

```
    Complex(int real = 0, int im = 0)
```

```
{
```

```
        this->real = real;
```

```
        this->im = im;
```

```
}
```

```
public:
```

```
    Complex operator + (const Complex & other)
```

```
{
```

```
    ...
```

```
    }  
};
```

Khởi tạo hai object Complex a(12, 7) và Complex b(5, 8). Điền vào ô ____ để khi chạy chương trình, $a + b$ là một phương thức hợp lệ nhưng a / b thì không

25. Cho đoạn code sau:

```
class A  
{  
private:  
    int data;  
    friend class B;  
};  
class B  
{  
protected:  
    int value;  
public:  
    A obj;  
};  
class C : public A, public B  
{  
  
};
```

Trong class C có thể truy cập được vào những thuộc tính nào của A và B?

- A. chỉ obj của B nhưng không thể gọi được obj.data để truy cập data của A
- B. value và obj của B nhưng không thể gọi được obj.data để truy cập data của A**
- C. value và obj của B, gọi được obj.data để truy cập data của A
- D. value và obj của B, data của A

26. Điều nào dưới đây là **sai** khi nói về destructor?

- A. Chỉ có thể khởi tạo 1 destructor duy nhất
- B. Destructor sẽ luôn được gọi khi chương trình kết thúc hoặc khi delete object được cấp phát bởi toán tử new
- C. Destructor có thể có tham số truyền vào**
- D. Destructor có ở cả struct và class

27. Cho đoạn code sau:

```
#include <iostream>
using namespace std;
class A
{
public:
    A()
    {
        cout << "Hello world!\n";
    }
    ~A()
    {
        cout << "Goodbye World!\n";
    }
};
class B : public A
{
public:
    B()
    {
        cout << "Hello\n";
    }
    ~B()
    {
        cout << "Goodbye!\n";
    }
};
int main()
{
    A* obj = new B;
    delete obj;
    return 0;
}
```

Sau lệnh `delete obj`, destructor của class nào sẽ được gọi?

- A. class A
- B. class B
- C. cả 2 class
- D. không có class nào

28. Cho đoạn code sau:

```
#include <iostream>
using namespace std;
class A
{
public:
    virtual void foo()
    {
        cout << "This is class A";
    }
};
class B : public A
{
public:
    void foo()
    {
        cout << "This is class B";
    }
};
int main()
{
    A* obj = new B;
    obj->foo();
    delete obj;
    return 0;
}
```

Kết quả khi chạy đoạn code trên là gì? This is class B

Điền câu trả lời vào đây:

29. Cho đoạn code sau:

```
class A
{
private:
    int data;
    friend class C;
};
class C
{
    A obj;
    friend class B;
```

```
};
class B : public A, public C
{
};
```

Ở hàm main ta thực hiện lệnh `B myObj`, khi đó `myObj` có thể truy cập được những thuộc tính nào của A và C?

- A. obj của C, không thể truy cập vào những thuộc tính của A kể cả khi gọi thông qua class C
- B. data của A, obj của C
- C. không thể truy cập được A và C
- D. có thể truy cập được mọi thuộc tính của A và C

30. Cho đoạn code sau:

```
struct Node
{
    int data;
    Node* next;
}

Node* addHead(Node* head)
{
    Node* tmp = new Node();
    tmp->data = data;
    tmp->next = nullptr;
    if(head == nullptr)
    {
        head = tmp;
    }
    else
    {
        _____;
        _____;
    }
}
```

```
    return head;
}
```

Thứ tự điền vào ô _____ (từ trên xuống) như thế nào là đúng để thêm 1 giá trị data vào đầu danh sách liên kết đơn (do_nothing có nghĩa là không điền vào)

- A. head = tmp, head->next = nullptr
- B. tmp->next = head, do_nothing
- C. head = tmp, do_nothing
- D. tmp->next = head, head = tmp

Tự luận: (2 câu 31, 32)

31. Một tòa nhà có cầu thang gồm N bậc, được đánh số từ 1 đến N (N là số tự nhiên, giới hạn $1 \leq N \leq 40$). Nền nhà được xem là bậc thứ 0. Mỗi lần lên cầu thang chỉ được lên 1 bậc hoặc tối đa là 2 bậc.

Hãy hiện thực hàm: `int countStep(int N)` để tính số cách leo lên cầu thang từ bậc thứ 0. Kết quả có thể rất lớn nên chỉ in số dư khi chia cho 1234577.

32. Do cầu thang đã cũ nên có k bậc bị hỏng, không leo lên được ($0 \leq k < N$, bậc thứ N không bị hỏng). Vị trí các bậc cầu thang hỏng (nếu có) được ghi lại bằng một mảng `breakStep` gồm k phần tử.

Hãy hiện thực hàm: `int countStepWithBreak(int N, int* breakStep, int k)` để đếm số cách leo lên cầu thang từ bậc thứ 0. Kết quả có thể rất lớn nên chỉ in số dư khi chia cho 1234577. (Nếu $k = 0$ thì `breakStep = nullptr`, khi đó sử dụng lại hàm `countStep` ở trên)